

This listing of claims will replace all prior versions, and listings of claims in the application:

Listing of Claims:

Claim 1 (original): A translation system for translating between nodes having heterogeneous file systems, comprising:

a consumer node having a first file system, the consumer node including a driver for supplementing requests from the first file system to a storage device;

an input/output (I/O) node implementing a second file system, the I/O node having the storage device connected thereto, the I/O node being in communication with the consumer node over a transport, the I/O node including,

a translator layer, the translator layer being configured to map the supplemented requests from the first file system to the second file system and back to the first file system.

Claim 2 (original): A translation system for translating between nodes having heterogeneous file systems as recited in claim 1, wherein the I/O node further comprises,

a message handler, the message handler being configured to interface with the driver so as to filter I/O requests to the translator layer.

Claim 3 (original): A translation system for translating between nodes having heterogeneous file systems as recited in claim 2, wherein the second file system is a dynamic flat file system, and the translator layer is interfaced between the dynamic flat file system and the message handler.

Claim 4 (original): A translation system for translating between nodes having heterogeneous file systems as recited in claim 1, wherein the translator layer is configured to reformat a mounted metadata of the second file system such that the reformatted metadata substantially matches the metadata of the first file system.

Claim 5 (original): A method for enabling communication between nodes having heterogeneous file systems, comprising:

generating a request to communicate with a desired I/O node that is connected to a nexus;

performing discovery of the desired I/O node;

enumerating the desired I/O node;

enumerating devices connected to the desired I/O node;

communicating a read request to a particular device of the enumerated devices associated with the desired I/O node;

intercepting the read request before communication over the nexus; and

supplementing the read request for communication over the nexus to the particular device that is connected to the desired I/O node.

Claim 6 (original): A method for enabling communication between nodes having heterogeneous file systems as recited in claim 5, wherein the performing of discovery of the desired I/O node comprises:

determining the type of the desired I/O node.

Claim 7 (original): A method for enabling communication between nodes having heterogeneous file systems as recited in claim 5, wherein the enumerating the desired I/O node comprises:

determining the characteristics of the desired I/O node.

Claim 8 (original): A method for enabling communication between nodes having heterogeneous file systems as recited in claim 5, wherein the enumerating devices connected to the desired I/O node comprises:

determining the type of the devices connected to the desired I/O node; and
determining the characteristics of the devices connected to the desired I/O node.

Claim 9 (original): A method for enabling communication between nodes having heterogeneous file systems as recited in claim 5, wherein the supplementing the read request for communication over the nexus to the particular device that is connected to the desired I/O node comprises:

manipulating the read request so as to create a supplemented request, the supplemented request being configured to carry a type of a file system of the node.

Claim 10 (original): A method for enabling communication between nodes having heterogeneous file systems, comprising:

receiving at an I/O node a request for communication from a consumer node;
determining file system type of the consumer node to be a first file system;
mounting a second file system at the I/O node;
loading metadata for the second file system at the I/O node;
reformatting the metadata of the second file system at the I/O node, the reformatting being performed to substantially match a metadata format of the first file system;
sending the reformatted metadata to the consumer node so that the consumer node can mount the received metadata; and

enabling communication between the consumer node and the I/O node, the reformatted metadata enabling transparent translation to and from the first file system and the second file system.

Claim 11 (original): A method for enabling communication between nodes having heterogeneous file systems as recited in claim 10, wherein the determining the file system type of consumer node to be a first file system, comprises:

receiving the supplemented message from a driver resident at the consumer node; and
analyzing the supplemented message by a translator layer so as to determine the type of the consumer node file system, the type of the consumer node file system being a first file system.

Claim 12 (original): A method for enabling communication between nodes having heterogeneous file systems as recited in claim 10, wherein the mounting a second file system at the I/O node comprises:

mapping the supplemented request of a first file system to a translated request of the second file system, the meaning of the translated request configured to be substantially equivalent to the meaning of the supplemented request.

Claim 13 (original): A method for enabling communication between nodes having heterogeneous file systems as recited in claim 10, wherein the I/O node can be one of a storage controller device, a device supporting NFS protocols, and a device supporting CIFS protocols, each of the devices for supporting NFS protocols and CIFS protocols being implemented for file system sharing.

Claim 14 (original): A method for enabling communication between nodes having heterogeneous file systems as recited in claim 10, wherein the consumer node can be one of a personal computer, a work station computer, a network computer, a file server, a computer server, a web server, a wireless computer, and a personal digital assistant.

Claim 15 (original): A method for enabling communication between nodes having heterogeneous file systems as recited in claim 10, wherein the second file system is a dynamic flat file system.

Claim 16 (original): A method for enabling communication between nodes having heterogeneous file systems, comprising:

generating a request to communicate with a desired I/O node that is connected to a nexus by a consumer node;

performing discovery and enumeration of the desired I/O node;

communicating an I/O request to a particular device of the enumerated devices associated with the desired I/O node;

supplementing the I/O request for communication over the nexus to the particular device that is connected to the desired I/O node;

receiving at an I/O node a request for communication from a consumer node;

determining file system type of the consumer node to be a first file system;

mounting a second file system at the I/O node;

loading metadata for the second file system at the I/O node;

reformatting the metadata of the second file system at the I/O node, the reformatting being performed to substantially match a metadata format of the first file system;

mounting using the reformatted metadata by the consumer node; and

enabling communication between the consumer node and the I/O node, the reformatted metadata enabling transparent translation to and from the first file system and the second file system.

Claim 17 (original): A method for enabling communication between nodes having heterogeneous file systems as recited in claim 16, wherein the performing discovery and enumeration of the desired I/O node comprises:

- performing discovery of the desired I/O node;
- enumerating the desired I/O node; and
- enumerating devices connected to the desired I/O node.

Claim 18 (original): A method for enabling communication between nodes having heterogeneous file systems as recited in claim 16, wherein the supplementing the I/O request for communication over the nexus to the particular device that is connected to the desired I/O node comprises:

- receiving the I/O request before communication over the nexus; and
- manipulating the I/O request so as to create a supplemented request, the supplemented request being configured to carry a type of a file system of the consumer node.

Claim 19 (original): A method for enabling communication between nodes having heterogeneous file systems as recited in claim 16, wherein the mount the reformatted data by the consumer node comprises:

- sending the reformatted metadata to the consumer node by the I/O node;
- loading the received reformatted metadata; and
- mounting using the loaded reformatted metadata.

Claim 20 (original): A method for enabling communication between nodes having heterogeneous file systems as recited in claim 16, wherein the second file system is a dynamic flat file system.